

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

Claim 1 (cancelled)

Claim 2 (currently amended): The method of claim ~~[[1]]~~29, wherein the ~~method~~ metadata further comprises a method handle.

Claim 3-4 (cancelled)

Claim 5 (currently amended): The method of claim ~~[[1]]~~29, further comprising storing the ~~method~~ metadata at an end of the instruction code ~~portion~~.

Claim 6 (cancelled)

Claim 7 (currently amended): The method of claim ~~[[1]]~~29, further comprising querying the instruction code ~~portion~~ for the ~~method~~ metadata using a maximum number of strides based on a maximum code size.

Claim 8-9 (cancelled)

Claim 10 (currently amended): The method of claim ~~[[1]]~~29, further comprising storing the ~~method~~ metadata between a first basic block and a second basic block of the instruction code ~~portion~~.

Claim 11 (currently amended): The method of claim ~~[[1]]~~29, further comprising using a compiler to store the ~~method~~ metadata in the instruction code ~~portion~~.

Claim 12 (currently amended): The method of claim 11, further comprising storing the ~~method~~ metadata in a basic block used for exception handling.

Claim 13-17 (cancelled)

Claim 18 (currently amended): An article comprising a machine-accessible storage medium containing instructions that when executed enable a system to:

store method metadata including a cookie indicator in a code portion of an instruction cache of a mobile platform device;

store the method metadata with a magic cookie having a bit pattern non-compliant with an instruction set architecture;

store the code portion, in a memory of the mobile platform device, for later use; and  
query the code portion for the method metadata with a maximum number of strides.

Claim 19 (cancelled):

Claim 20 (previously presented): The article of claim 18, further comprising instructions that when executed enable the system to store the method metadata at an N-aligned address of the code portion, wherein  $N=2^x$  and x is an integer.

Claim 21 (cancelled):

Claim 22 (previously presented): The article of claim 18, further comprising instructions that when executed enable the system to query the code portion at N-aligned addresses, wherein  $N=2^x$  and x is an integer.

Claim 23 (previously presented): The article of claim 18, further comprising further comprising instructions that when executed enable the system to store the method metadata between a first basic block and a second basic block of the code portion.

Claim 24-28 (cancelled)

Claim 29 (new) A method comprising:

locating compiled instruction code in an instruction cache of a mobile platform device, the instruction code including a first N-aligned address, a second N-aligned address located immediately after the first N-aligned address, and a third N-aligned address located immediately after the second N-aligned address; wherein  $N=2^x$  and x is an integer;

storing metadata at one of a first location, a second location, or a third location; the first location being based on the first N-aligned address, the second location being based on the second N-aligned address, the third location being based on the third N-aligned address, wherein the metadata comprises a bundle that includes a cookie and method information, the bundle having a first bundle size and the cookie having a bit pattern non-compliant with an instruction set architecture;

confirming an instruction pointer points to a valid address included in the instruction code;

searching for the metadata at the first location;

if the metadata is still not located, searching for the metadata at the second location; and

if the metadata is still not located, searching for the metadata at the third location;

wherein, to expedite the searching process, searching for the metadata is limited to searching only in the instruction cache.

Claim 30 (new) The method of claim 29, further comprising searching for the metadata at the first location; wherein the first location is located immediately prior to the first N-aligned address.

Claim 31 (new) The method of claim 30, further comprising searching for the metadata at the first location; wherein the first location is located a first distance behind the first N-aligned address and the first distance is equal to the first bundle size.

Claim 32 (new) The method of claim 29, further comprising searching for the metadata at the first location; wherein the first location is located immediately after to the first N-aligned address.

**Claim 33 (new)**      The method of claim 29, further comprising searching for the metadata in an alignment padding space located between a first block and a second block, the alignment padding space including the first location and a literal pool.

**Claim 34 (new)**      The method of claim 29, further comprising confirming, via a compiler, that a first block will be queried frequently and storing the metadata in the first block.